

Session 3 Part 1

Use of Arrays to save multiple values.

Aims of this section:

To discuss the use of arrays in programming, in particular their operation.

To work through some array examples on the whiteboard with a focus on:

- How to set-up an array.
- Adding values to an array.
- Removing values from an array.
- Totally and averaging the values in an array.

To work through some array examples on the micro:bit to:

- Initialise an array.
- Adding values to an array, that are coming in via radio communication.
- Sending these values over the serial port, Downloading the CSV for them.
- Demonstrate a simple graph of data in excel.
- Totally and averaging the values in an array on the micro:bit.



TASK

Assume : Light level is being sent via radio signal from one microbit to another.

AIM:

Take the light level received and store it in an array.

When button A pressed the array data is analysed to do the following:

- *Get the total of all the values in the array*
- *Get the average value for that day*

When Button B is pressed all data in the array is:

- *Displayed*
- *Sent via the serial port.*

Pre- Coding Considerations for Arrays.

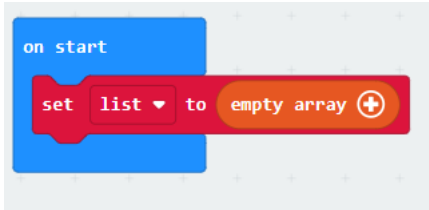
Embed Video here

Use of Arrays to save multiple values.

Next Step is to code this on the micro:bit.

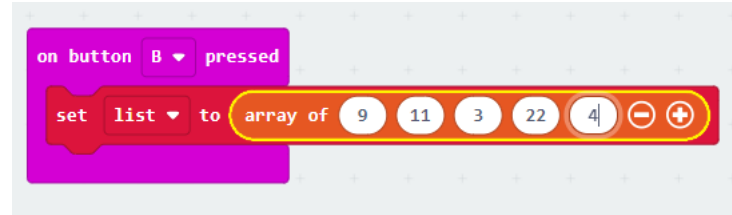
<https://makecode.com/multi#>

Initialise Array



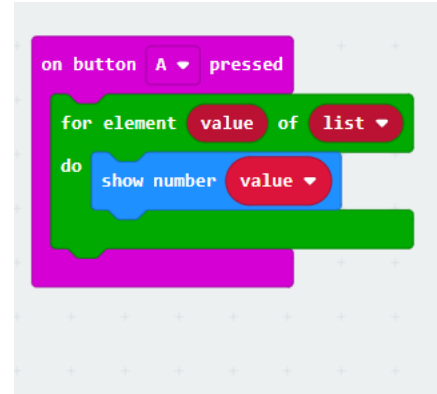
```
on start
  set list to empty array
```

Set values in array



```
on button B pressed
  set list to array of 9 11 3 22 4
```

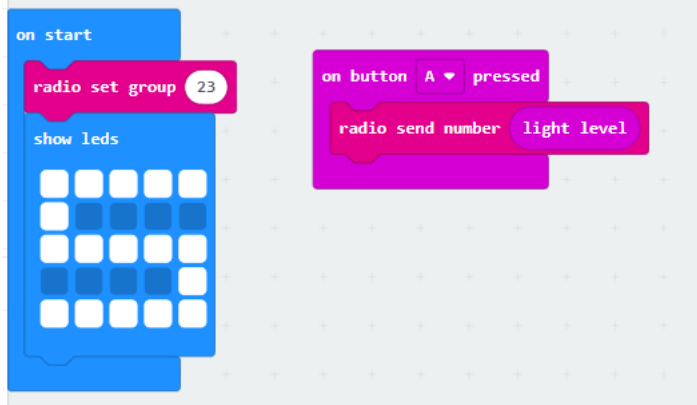
Loop through all values in Array



```
on button A pressed
  for element value of list
  do
    show number value
```

Use of Arrays to save multiple values.

Light Sender Code



This code has two parts:

OnStart – Here the radio channel is set and an S is displayed so when powered it is easy to see that this micro:bit is the sender.

OnButtonAPressed – Here the light level from the sender micro:bit is sent to the receiver.

Use of Arrays to save multiple values.

This is the Sender Code.

This is explained on the next slide.

```
on start
  serial redirect to USB
  radio set group 23
  show leds
  set LightList to empty array

on button A pressed
  set Total to 0
  for index from 0 to length of array LightList - 1
  do
    set Total to Total + LightList get value at index
  show number Total
  change Average by Total ÷ length of array LightList
  show number Average

on radio received receivedNumber
  LightList add value receivedNumber to end

on button B pressed
  for element value of LightList
  do
    serial write line value
    show number value
```

Use of Arrays to save multiple values.

Receiver Code with Array and Serial Explanation:

OnStart – Ensure data can be sent via the USB, Set radio channel to 23, show the letter R to indicate that this micro:bit is the receiver and finally initialise the array

to an empty array.

OnRadioReceived- This receives the light level from the sender and adds the value to the end of the array.

OnButtonAPressed – In this part we are looping through the array and getting the total and the average for the values in the array.

OnButtonBPressed - This part loops through the array and sends all the values along the serial port. This can be used to generate a CSV file or to be picked up by another application like Python or be sent to storage in a database.

We are also showing each value after it is sent but this is not essential it is just a debugging part of the code to ensure that the values are processed.

Arrays



Any Questions?

Stretch break



Extra Slide with Bubble Sort

Not covered on the day but discussed as an extension to the array code. This will sort the array therefore allowing students to do further analysis e.g. mode and median.

This is tricky code so designing out the solution beforehand is always of benefit.

```

on start
  set myList to array of 7 1 3 9 4

function bubbleSort
  repeat length of array myList times
    do
      set index to 0
      repeat length of array myList - 1 times
        do
          if <myList get value at index > <myList get value at index + 1 > then
            set tempValue to myList get value at index
            myList set value at index to myList get value at index + 1
            myList set value at index + 1 to tempValue
          change index by 1
        do
      end repeat
    do
  end repeat

on button A pressed
  call bubbleSort
  for element value of myList
    do
      show number value
    do
  end for

```